UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/729,684	12/05/2003	Michael Hong	252209-1020	3198

24504 7590 06/03/2008
THOMAS, KAYDEN, HORSTEMEYER & RISLEY, LLP
600 GALLERIA PARKWAY, S.E.
STE 1500
ATLANTA, GA 30339-5994

EXAMINER	
HSU, JONI	

ART UNIT	PAPER NUMBER
2628	

MAIL DATE	DELIVERY MODE
06/03/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/729,684	HONG ET AL.	
	Examiner	**Art Unit**	
	JONI HSU	2628	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *18 March 2008*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-10,12-23 and 25-27* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-10,12-23 and 25-27* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date *2/18/08*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Information Disclosure Statement

1. Information disclosure statement (IDS) submitted on February 18, 2008 was filed after mailing date of application on December 5, 2003. Submission is in compliance with provisions of 37 CFR 1.97. Accordingly, information disclosure statement is being considered by examiner.

Response to Arguments

2. Applicants arguments, filed on March 18, 2008, are considered but are not persuasive.

3. Applicant argues "levels" disclosed in Voorhies (US007023437B1) relate to different levels in z-pyramid and not to two-level z-test (p. 14-15).

In reply, Examiner points out "levels" does relate to different levels in z-pyramid, however, Fig. 11 in Voorhies shows flowchart of z-test, and shows z-test (1114) is first performed on one level (c. 16, ll. 5-8; c. 6, ll. 28-29). If that level is not finest level (1118), z-testing is then processed for next finest level (c. 16, ll. 1-22; c. 6, ll. 41-45). Since z-test is performed for each level in z-pyramid, each of these z-tests associated with each of these levels in z-pyramid is considered to be a level in z-test, and so there are plurality of levels in z-test, and so this is considered to be two-level z-test.

4. Applicant argues Voorhies does not teach second level z-test is performed only on pixels within record of compressed z-information in which first level z-test determines that some but not all pixels of associated macropixel are visible (p. 15-16).

In reply, Examiner points out Voorhies describes "control proceeds to step 1114, where if polygon's status is visible, the procedure terminates at step 1116, and otherwise, control returns to step 1106, and otherwise, control returns to step 1106" (c. 16, ll. 5-8). So, if 1st level z-test

(1114) determines that all pixels of associated macropixel are visible, then procedure terminates, and 2nd level z-test is not performed (c. 16, ll. 5-8). If 1st level z-test determines that some but not all pixels of associated macropixel are visible, then procedure returns to step 1106 so that 2nd level z-test can be performed (c. 16, ll. 1-22; c. 6, ll. 41-45). Since procedure only continues on to 2nd level z-test if 1st level z-test determines that some but not all pixels of associated macropixel are visible, this means 2nd level z-test is performed only on pixels within record of z-information in which 1st level z-test determines that some but not all pixels of associated macropixel are visible. Orenstein (US006580427B1) is used to teach compressed z-information.

5. Applicant argues neither Voorhies nor Orenstein teaches "creating a visibility mask for each primitive, wherein for each primitive, the visibility mask indicates whether the primitive is clipped, culled, or is a zero-pixel primitive." Griffin fails to disclose "zero-pixels" (p. 17-18).

In reply, Examiner points out Voorhies teaches creating visibility mask for each primitive (c. 33, ll. 42-47), wherein for each primitive, visibility mask indicates whether primitive is clipped or culled (c. 30, ll. 50-54; c. 3, ll. 49-55; c. 12, ll. 37-42). According to Applicant's disclosure, zero-pixel primitive is primitive that, when rendered, consumes less area than one pixel of visibility [0024]. Griffin teaches detecting when polygon only partially covers pixel region (c. 2, ll. 61-c. 3, ll. 5; c. 5, ll. 26-42), and so teaches detecting when rendered polygon consumes less area than one pixel of visibility for pixel region, and so teaches detecting zero-pixel primitives. So, Griffin teaches visibility mask indicates whether primitive is zero-pixel primitive (c. 34, ll. 56-58; c. 2, ll. 61-c. 3, ll. 5; c. 5, ll. 26-42).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

8. Claims 1, 4, 5, 7, and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Voorhies (US007023437B1) and Orenstein (US00658042B1).

9. As per Claim 1, Voorhies teaches multi-pass method of rendering plurality of graphic primitives (c. 2, ll. 58-67; c. 6, ll. 28-29) comprising in 1st pass: passing only limited portion of graphic data for each primitive through graphic pipeline, limited portion of graphic data has location-related data (c. 3, ll. 16-31). According to Applicant's disclosure, compressed z-buffer effectively provides condensed depth information for multiple pixels, such that grouping of pixels (or macro-pixel) may by trivially accepted if all pixels of current macro-pixel are deemed to be in front of previously-stored pixels or trivially rejected if all pixels of current macro-pixel primitive are deemed to be behind previously-stored pixels [0023]. Voorhies teaches macro-pixel (16x16 pixel region) may be trivially accepted if all pixels of current macro-pixel are deemed to

be in front of previously-stored pixels or trivially rejected if all pixels of current macro-pixel primitive are deemed to be behind previously-stored pixels (c. 54, ll. 44-55; c. 6, ll. 1-14). Z-pyramid data structure is hierarchical z-buffer (c. 33, ll. 36-39; c. 55, ll. 66-67). So, Voorhies teaches processing limited portion of graphic data to build z-buffer, z-buffer having plurality of z-records, each z-record embodying z information for plurality of pixels (c. 54, ll. 44-55; c. 6, ll. 1-14). Record for each fragment includes coverage mask indicating image samples covered by fragment, and this record format is designed to resolve visibility at each image sample (c. 33, ll. 42-49). So bits on coverage mask are set to indicate whether image samples in primitive are visible or not, this is considered to be setting visibility indicator, for each primitive, if any pixel of primitive is determined to be visible. In 2nd pass: for each primitive, determining whether associated visibility indicator for that primitive is set; discarding, without passing through graphic pipeline, primitives for which associated visibility indicator is not set; passing remaining portion of graphic data for each primitive determined to have associated visibility indicator set (c. 54, ll. 44-55; c. 46, ll. 61-c. 47, ll. 4; c. 3, ll. 16-35). Multi-level z test is performed, and test continues to proceed to another level until finest level is reached (c. 16, ll. 1-22). Fig. 11 shows flowchart of z-test, and shows z-test (1114) is first performed on one level, this includes comparing graphic data of current primitive with corresponding information in z-buffer (c. 16, ll. 5-8; c. 6, ll. 28-29), this is considered to be 1st level of the z-test. If that level is not finest level (1118), z-testing is then processed for next finest level, and z-test continues to proceed until finest level is reached, z-testing at finest level is performed on per-pixel basis in z-test manner (c. 16, ll. 1-22; c. 6, ll. 41-45), this is considered to be 2nd level of the z-test. 2nd level of z-test depends in part on outcome of 1st level z-test (c. 54, ll. 44-55). So, Voorhies teaches performing

two-level z test on graphic data, 1st level of z-test compares graphic data of current primitive with corresponding information in z-buffer, 2nd level of z-test is performed on per-pixel basis in z-test manner, 2nd level z-test is performed only on pixels within record of z-information in which 1st level z-test determines some but not all pixels of associated macropixel are visible (c. 54, ll. 44-55; c. 6, ll. 39-43; c. 16, ll. 1-22; c. 6, ll. 28-29, 41-45). Visible geometry is rendered (c. 5, ll. 50-53), rendering includes shading (c. 6, ll. 15-17). So Voorhies teaches communicating data associated with pixels of macropixels determined to be visible to pixel shader for rendering. “Levels” does relate to different levels in z-pyramid, however, Fig. 11 shows flowchart of z-test, and shows z-test (1114) is first performed on one level (c. 16, ll. 5-8; c. 6, ll. 28-29). If that level is not finest level (1118), z-testing is then processed for next finest level (c. 16, ll. 1-22; c. 6, ll. 41-45). Since z-test is performed for each level in z-pyramid, each of these z-tests associated with each of these levels in z-pyramid is considered to be a level in z-test, and so there are plurality of levels in z-test, and so this is considered to be two-level z-test. Voorhies describes "control proceeds to step 1114, where if polygon's status is visible, the procedure terminates at step 1116, and otherwise, control returns to step 1106, and otherwise, control returns to step 1106” (c. 16, ll. 5-8). So, if 1st level z-test (1114) determines that all pixels of associated macropixel are visible, then procedure terminates, and 2nd level z-test is not performed (c. 16, ll. 5-8). If 1st level z-test determines that some but not all pixels of associated macropixel are visible, then procedure returns to step 1106 so that 2nd level z-test can be performed (c. 16, ll. 1-22; c. 6, ll. 41-45). Since procedure only continues on to 2nd level z-test if 1st level z-test determines that some but not all pixels of associated macropixel are visible, this means 2nd level

z-test is performed only on pixels within record of z-information in which 1st level z-test determines that some but not all pixels of associated macropixel are visible.

However, Voorhies does not explicitly teach z-buffer is compressed z-buffer, compressed z-buffer comprising plurality of z-records, each z-record embodying z information for plurality of pixels such that condensed depth information for plurality of pixels is represented by single z-record. However, Orenstein teaches z-buffer is compressed z-buffer, compressed z-buffer comprising plurality of z-records (blocks), each z-record embodying z information for plurality of pixels such that condensed depth information for plurality of pixels is represented by single z-record (c. 4, ll. 50-65; c. 9, ll. 43-45).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify Voorhies so z-buffer is compressed z-buffer comprising z-records, each z-record embodying z information for plurality of pixels such that condensed depth information for plurality of pixels is represented by single z-record because Orenstein suggests significant amounts of z-data are transferred between memory and graphics resources during rendering stage which place significant burdens on bandwidth of memory channel, and consequent reduction in memory bandwidth reduces performance of graphics system, so it is advantageous to use compressed z-buffer containing compressed z-data that may be transferred with significantly lower impact on bandwidth of memory channel (c. 2, ll. 13-30; c. 4, ll. 11-22, 58-65).

10. As per Claim 4, Voorhies teaches each z-record (c. 54, ll. 44-55) has minimum z value for pixels, maximum z values for pixels (c. 8, ll. 43-55), and coverage mask indicating which of the pixels are visible for current primitive (c. 33, ll. 42-47; c. 6, ll. 28-29).

However, Voorhies does not explicitly teach that the z-record is compressed. However, Orenstein teaches this limitation, as discussed in the rejection for Claim 1.

11. As per Claim 5, Voorhies teaches each z-record (c. 54, ll. 44-55) has 2 minimum z values for pixels, 2 maximum z values for pixels (c. 8, ll. 43-55), coverage mask indicating which pixels are visible for current primitive (c. 33, ll. 42-47; c. 6, ll. 28-29).

However, Voorhies does not explicitly teach that the z-record is compressed. However, Orenstein teaches this limitation, as discussed in the rejection for Claim 1.

12. As per Claim 7, parser is known in the art to be component of compiler that forms data structure, usually a tree, that is suitable for later processing and captures implied hierarchy of input. Voorhies teaches parser forms tree data structure that is suitable for later processing and captures hierarchy of input (c. 9, ll. 57-61), discarding is performed by parser (c. 54, ll. 44-55).

13. As per Claim 13, Voorhies teaches method of rendering plurality of graphic primitives (c. 2, ll. 58-67; c. 6, ll. 28-29) comprising passing in 1st pass, in graphic pipeline, only limited portion of graphic data for each primitive, each primitive has plurality of pixels and limited portion of graphic data has location-related data (c. 3, ll. 16-31; c. 6, ll. 40-44, c. 6, ll. 66-67-c. 7, ll. 3); processing limited portion of graphic data to build z-buffer, z-buffer having plurality of z-records, each z-record embodying z information for plurality of pixels (c. 54, ll. 44-55; c. 6, ll. 1-14; c. 33, ll. 36-39; c. 55, ll. 66-67); in 2nd pass, within graphic pipeline, performing two-level z-test on graphic data, wherein first level of z-test compares graphic data of current primitive with corresponding information in z-buffer, and 2nd level of z-test is performed on per-pixel basis in z-test manner, 2nd level z-test is performed only on pixels within record of z-information in which 1st level z-test determines some but not all pixels of macropixel (16x16 pixel region) are visible,

additional graphic data associated with each primitive is passed into graphics pipeline on 2^{nd} pass only for primitives that are at least partially visible (c. 54, ll. 44-55; c. 6, ll. 39-43; c. 16, ll. 1-22; c. 3, ll. 16-35, Fig. 11); and communicating data associated with pixels of macropixels determined to be visible to pixel shader for rendering (c. 5, ll. 50-53; c. 6, ll. 15-17).

However, Voorhies does not explicitly teach z-buffer is compressed z-buffer, compressed z-buffer comprising plurality of z-records, each z-record embodying z information for plurality of pixels such that condensed depth information for plurality of pixels is represented by single z-record. However, Orenstein teaches this, as discussed in the rejection for Claim 1.

14. Claims 3 and 6 are rejected under 35 U.S.C. 103(a) as being unpatentable over Voorhies (US007023437B1) and Orenstein (US006580427B1) in view of Gannett (US006118452A).

15. As per Claim 3, Voorhies and Orenstein are relied upon for teachings discussed for to Claim 1. Voorhies teaches location-related data comprises X, Y, and Z values (c. 14, ll. 22-38).

However, Voorhies and Orenstein do not teach location-related data has W values. But, Gannett teaches location-related data has X, Y, Z and W values (c. 1, ll. 29-33; c. 13, ll. 50-55).

It would have been obvious to one of ordinary skill in the art at time of invention by applicant to modify Voorhies and Orenstein so location-related data has W values because Gannett teaches W is needed in order to determine horizontal length of pixels to render, and W is commonly used in typical computer graphics systems (c. 1, ll. 18-33; c. 13, ll. 50-55).

16. As per Claim 6, Voorhies does not teach setting visibility indicator is setting bit in frame buffer memory. However, Gannett teaches this limitation (c. 13, ll. 16-19; c. 14, ll. 13-22).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify Voorhies so setting visibility indicator comprises setting bit in frame buffer

because Gannett suggests setting bits in mask is quick and efficient way to indicate visibility (c. 13, ll. 16-19; c. 14, ll. 13-22).

17. Claims 8, 9, 12, 14-17, 19-23, and 25-27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Voorhies (US007023437B1) and Orenstein (US006580427B1) in view of Griffin (US005990904A).

18. As per Claim 8, Voorhies teaches rendering plurality of graphic primitives comprising passing, within graphic pipeline, only limited portion of graphic data associated with each primitive, limited portion of graphic data has location-related data (c. 2, ll. 58-67; c. 6, ll. 28-29; c. 3, ll. 16-31); each primitive has plurality of pixels (c. 6, ll. 40-44, c. 6, ll. 66-67-c. 7, ll. 3); processing limited portion of graphic data associated with each individual primitive to build z-buffer for each primitive, each z-buffer has z-information for macro-pixel; creating visibility mask for each primitive (c. 33, ll. 42-47), wherein for each primitive, visibility mask indicates whether primitive is clipped or culled (c. 30, ll. 50-54; c. 3, ll. 49-55; c. 12, ll. 37-42); determining, for each primitive, whether primitive has at least 1 visible pixel based on visibility mask (c. 33, ll. 42-47); communicating data associated with pixels of primitives determined to have at least 1 visible primitive to pixel shader for rendering; passing, processing, within pixel shader, remaining graphic data associated with each primitive only for those primitives determined to have at least one visible pixel, remaining graphic data includes at least one of following: lighting, texture, fog data (c. 8, ll. 50-58; c. 54, ll. 44-55; c. 6, ll. 1-17, 39-43; c. 16, ll. 1-22; c. 3, ll. 16-35; c. 5, ll. 50-53; c. 33, ll. 36-39; c. 55, ll. 66-67).

But, Voorhies does not explicitly teach z-buffer is compressed z-buffer, each compressed z-buffer contains plurality of z-records which each contain compressed z-information for macro-

pixel. But, Orenstein teaches compressed z-buffer contains plurality of z-records (block) which each contain compressed z-information for macro-pixel (4x4 array of pixels (span)) (c. 4, ll. 50-65; c. 9, ll. 43-45). Since Voorhies teaches building z-buffer for each primitive (c. 8, ll. 50-58), z-buffers of Voorhies can be modified so they are compressed z-buffers, as suggested by Orenstein. This would be obvious for reasons for Claim 1.

But, Voorhies and Orenstein do not teach visibility mask indicates whether primitive is zero-pixel primitive. According to Applicant's disclosure, zero-pixel primitive is primitive that, when rendered, consumes less area than one pixel of visibility [0024]. Griffin teaches detecting when polygon only partially covers pixel region (c. 2, ll. 61-c. 3, ll. 5; c. 5, ll. 26-42), so teaches detecting when rendered polygon consumes less area than one pixel of visibility for pixel region, and so teaches detecting zero-pixel primitives. So, Griffin teaches visibility mask indicates whether primitive is zero-pixel primitive (c. 34, ll. 56-58; c. 2, ll. 61-c. 3, ll. 5; c. 5, ll. 26-42).

It would have been obvious to one of ordinary skill in the art at the time of invention by applicant to modify Voorhies and Orenstein so visibility mask indicates whether primitive is zero-pixel primitive because Griffin teaches being able to perform anti-aliasing so anomalies such as jaggy edges in rendered image do not result (c. 2, ll. 61-c. 3, ll. 5). It would be obvious to include compressed z-buffer because Griffin teaches considerably reducing amount of data required, allowing implementation of more sophisticated anti-aliasing algorithm (c. 9, ll. 34-54).

19. As per Claim 9, Voorhies teaches setting visibility indicator for each pixel determined to have at least one visible pixel (c. 33, ll. 42-49).

20. As per Claim 12, Voorhies teaches determining whether primitive has at least one visible pixel ensures primitive does not fail z-buffer test (c. 54, ll. 44-55; c. 6, ll. 1-14), ensures all pixels

of primitive are not culled (c. 3, ll. 49-55), and ensures all pixels of primitive are not clipped (c. 12, ll. 37-42).

However, Voorhies does not teach z-buffer is compressed z-buffer. However, Orenstein teaches this limitation, as discussed in rejection for Claim 1.

But, Voorhies, Orenstein do not teach ensuring primitive does not render to zero pixels. According to Applicant's disclosure, zero-pixel primitive is primitive that, when rendered, consumes less area than one pixel of visibility [0024]. Griffin teaches compressed z-buffer (c. 9, ll. 34-54), ensuring primitive does not render to zero pixels (c. 2, ll. 61-c. 3, ll. 5; c. 5, ll. 26-42). This would be obvious for reasons for Claim 8.

21. As per Claim 14, Voorhies teaches graphics processor having 1st-pass logic that delivers to graphic pipeline, in 1st pass, only limited portion of graphic data for each primitive, each primitive has plurality of pixels, limited portion of graphic data has location-related data (c. 3, ll. 16-31; c. 6, ll. 40-44, c. 6, ll. 66-67-c. 7, ll. 3); logic that processes limited portion of graphic data for each primitive to create z-buffer having plurality of z-records, z-information for macro-block is placed into each of plurality of z-records (c. 54, ll. 44-55; c. 6, ll. 1-14; c. 33, ll. 36-39; c. 55, ll. 66-67); logic configured to create visibility mask for each primitive (c. 33, ll. 42-47), wherein for each primitive, visibility mask indicates whether primitive is clipped or culled (c. 30, ll. 50-54; c. 3, ll. 49-55; c. 12, ll. 37-42); logic that determines, for each primitive, whether primitive has at least 1 visible pixel based on visibility mask (c. 33, ll. 42-47; c. 54, ll. 44-55; c. 6, ll. 1-14); 2nd-pass logic that delivers to graphic pipeline, in 2nd pass, remaining graphic data associated with each primitive for only those primitives determined to have visible pixel, 2nd-

pass logic configured to inhibit delivery of graphic data to graphic pipeline for primitives not determined to have visible pixel (c. 54, ll. 44-55; c. 6, ll. 39-43; c. 16, ll. 1-22; c. 3, ll. 16-35).

But, Voorhies does not explicitly teach z-buffer is compressed z-buffer comprising plurality of z-records, z-information for macro-block is compressed into each of plurality of z-records such that condensed depth information for macro-block is represented by single z-record. But, Orenstein teaches z-buffer is compressed z-buffer comprising plurality of z-records (blocks), z-information for macro-block (4x4 array of pixels (span)) is compressed into each of plurality of z-records such that condensed depth information for macro-block is represented by single z-record (c. 4, ll. 50-65; c. 9, ll. 43-45). This would be obvious for reasons for Claim 1.

But, Voorhies and Orenstein do not teach visibility mask indicates whether primitive is zero-pixel primitive. But, Griffin teaches this limitation, as discussed in rejection for Claim 8.

22. As per Claim 15, parser is known in the art to be component of compiler that forms data structure, usually tree, which is suitable for later processing and which captures implied hierarchy of input. Voorhies teaches parser that forms tree data structure which is suitable for later processing and which captures hierarchy of input (c. 9, ll. 57-61), and first-pass logic and second-pass logic are contained within parser (c. 54, ll. 44-55).

23. As per Claim 16, it is similar in scope to Claim 12, so is rejected under same rationale.

24. As per Claim 17, Voorhies discloses including logic for setting a visibility indicator for each primitive determined to have at least one visible pixel (c. 33, ll. 42-49).

25. As per Claim 19, Voorhies teaches including logic configured to associate each primitive processed in the first pass of the data with a distinct visibility indicator (c. 33, ll. 42-49).

26. As per Claim 20, Voorhies discloses including logic configured to evaluate, for each primitive presented for processing in the second pass, a status of the visibility indicator associated with the given primitive (c. 54, ll. 44-55; c. 46, ll. 61-c. 47, ll. 4).

27. As per Claim 21, Voorhies teaches graphics processor having logic that passes and processes only portion of graphic data passed into graphic pipeline for each of plurality of primitives, in 1st pass within graphic pipeline to determine whether primitive has at least one visible pixel, wherein each primitive comprises plurality of pixels, and graphic data has location-related data (c. 3, ll. 16-31; c. 6, ll. 1-14, 40-44, c. 6, ll. 66-67-c. 7, ll. 3; c. 54, ll. 44-55); logic building z-buffer from processing of graphic data in first pass, z-buffer comprising plurality of z-records, z-information for a macro-block is placed into single record (c. 54, ll. 44-55; c. 6, ll. 1-14; c. 33, ll. 36-39; c. 55, ll. 66-67); logic creating visibility mask for each primitive (c. 33, ll. 42-47), for each primitive, visibility mask indicates whether primitive is clipped or culled (c. 30, ll. 50-54; c. 3, ll. 49-55; c. 12, ll. 37-42); logic that renders, in 2nd pass within graphic pipeline, only primitives determined in 1st pass to have at least one visible pixel based on visibility mask, remaining portion of graphic data associated with each primitive is passed into graphics pipeline on 2nd pass (c. 33, ll. 42-47; c. 54, ll. 44-55; c. 6, ll. 39-43; c. 16, ll. 1-22; c. 3, ll. 16-35).

However, Voorhies does not explicitly teach z-buffer is compressed z-buffer comprising plurality of z-records, wherein z-information for macro-block is compressed into single record such that condensed depth information for macro-block is represented by single record. However, Orenstein teaches this limitation, as discussed in the rejection for Claim 14.

However, Voorhies and Orenstein do not teach visibility mask indicates whether primitive is zero-pixel primitive. But, Griffin teaches this limitation, as discussed for Claim 8.

28. As per Claim 22, it is similar in scope to Claim 12, so is rejected under same rationale.

29. As per Claim 23, parser is known in the art to be component of compiler that forms data structure, usually tree, which is suitable for later processing and which captures the implied hierarchy of input. Voorhies teaches parser that forms tree data structure which is suitable for later processing and which captures hierarchy of input (c. 9, ll. 57-61), and logic configured to limit processing of graphic data is within parser (c. 54, ll. 44-55).

30. As per Claim 25, Voorhies discloses including logic for setting a visibility indicator for each primitive processed in the first pass (c. 33, ll. 42-49).

31. As per Claim 26, Voorhies teaches logic evaluating visibility indicator for each primitive prior to submitting primitive to logic rendering in 2^{nd} pass (c. 54, ll. 44-55; c. 46, ll. 61-c. 47, ll. 4).

32. As per Claim 27, it is similar in scope to Claim 5, so is rejected under same rationale.

33. Claims 10 and 18 are rejected under 35 U.S.C. 103(a) as being unpatentable over Voorhies (US007023437B1), Orenstein (US0065804427B1), and Griffin (US005990904A) in view of Gannett (US006118452A).

Claims 10 and 18 are similar in scope to Claim 6, so are rejected under same rationale.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO

MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of advisory action. In no event, however, will statutory period for reply expire later than SIX MONTHS from date of final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joni Hsu whose telephone number is 571-272-7785. The examiner can normally be reached on M-F 8am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kee Tung can be reached on 571-272-7794. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

JH

/Kee M Tung/
Supervisory Patent Examiner, Art Unit 2628